# Block diagram

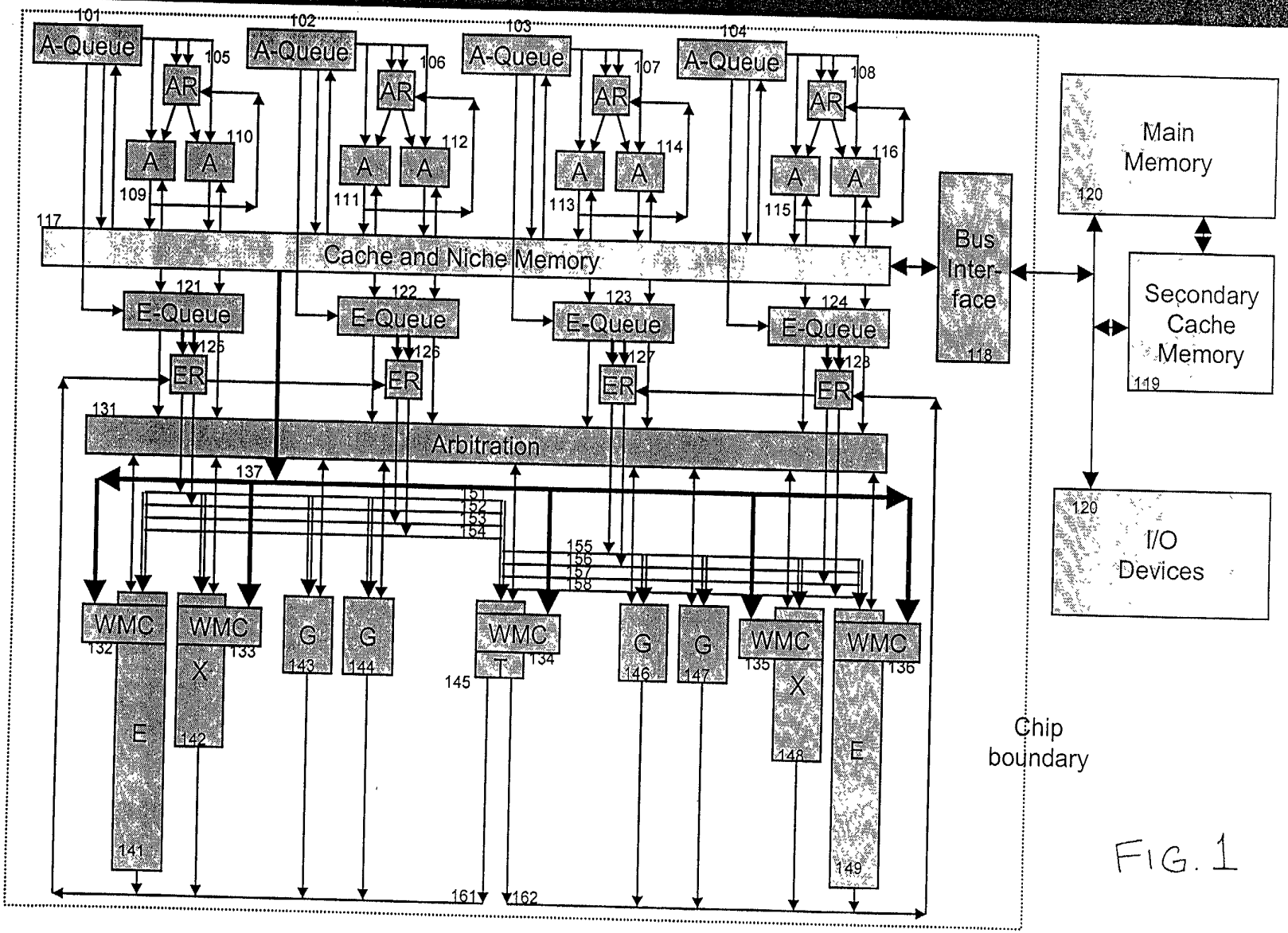



FIG. 1

# Wide multiply matrix

- $rd_{128} = m[rc]_{(128*64/size)} * rb_{128}$

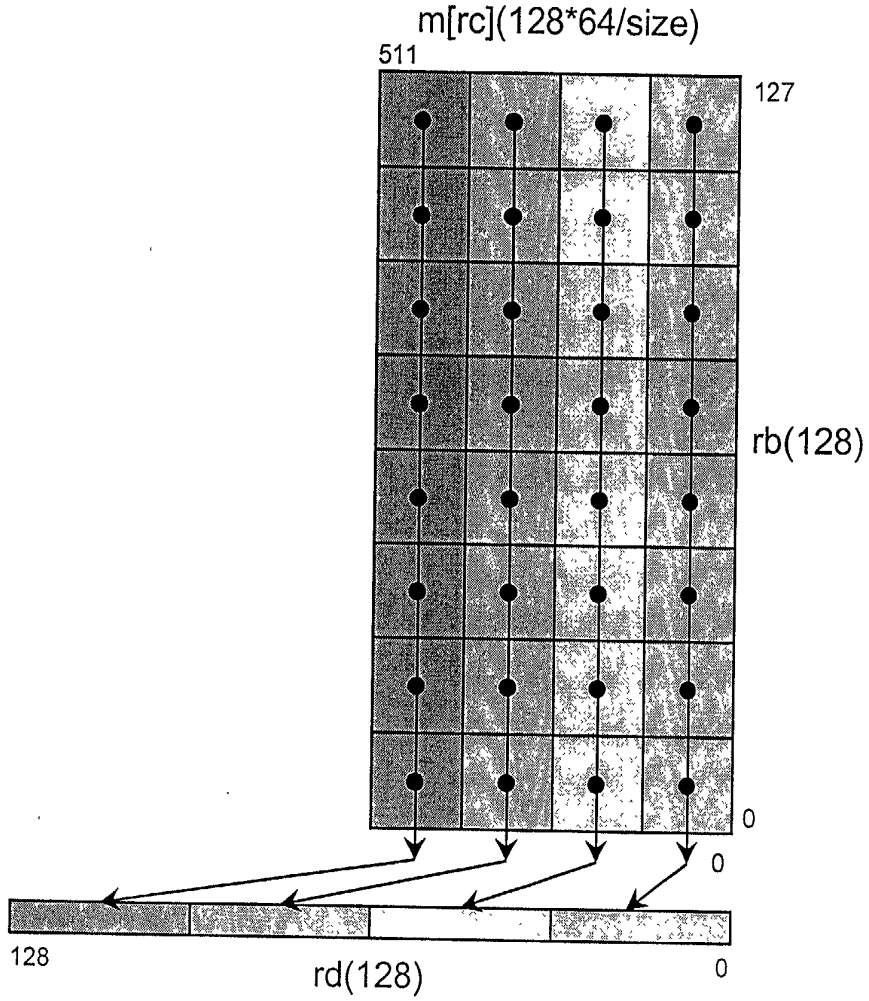



FIG. 2

# Wide multiply matrix

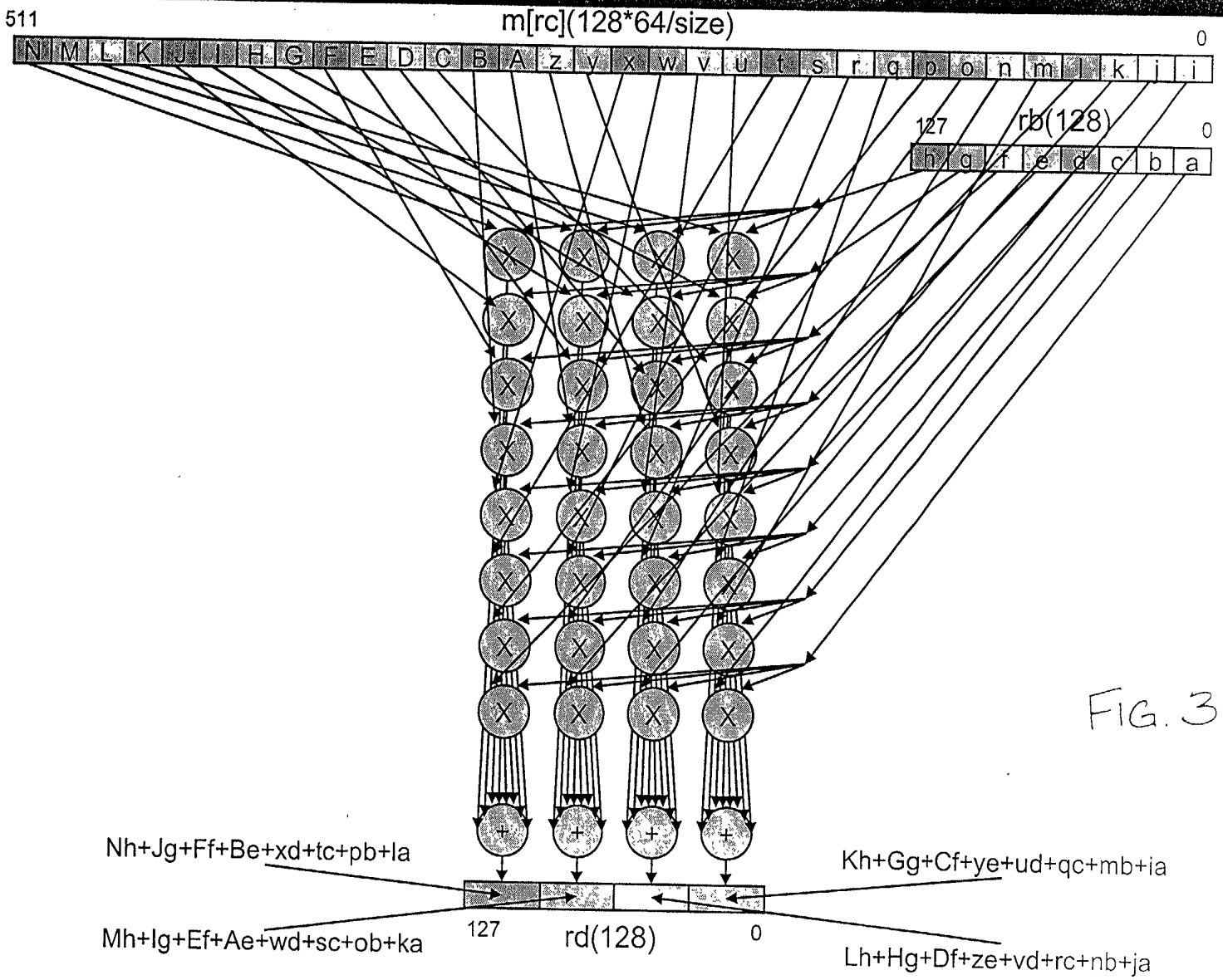



FIG. 3

# SMT + DAE

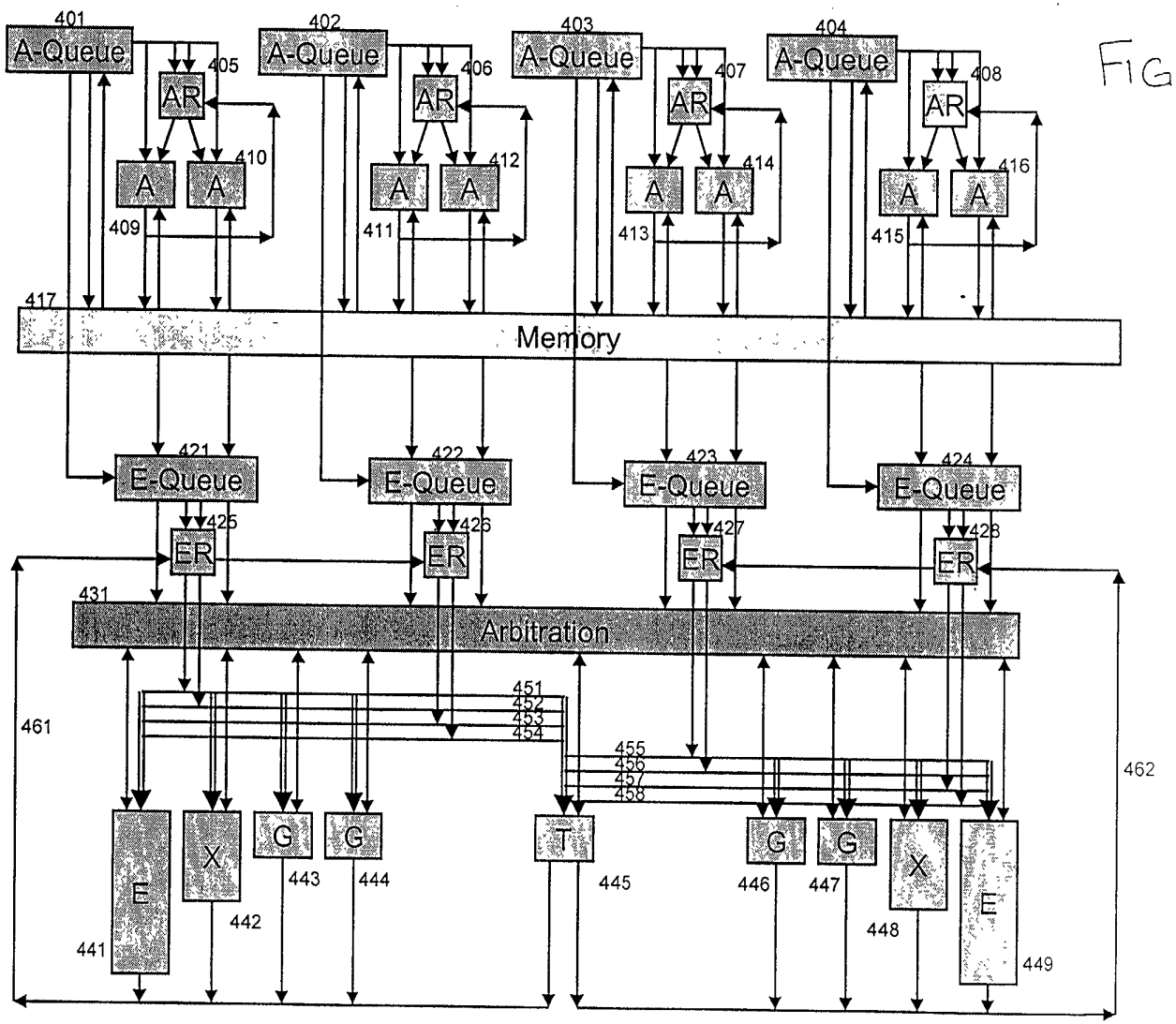
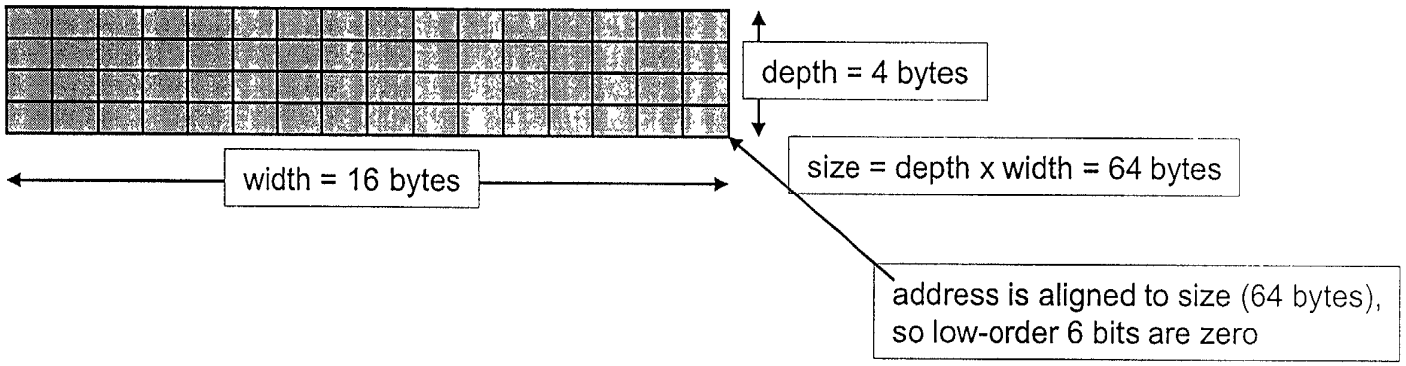

401
A-Queue
402
A-Queue
403
A-Queue
404
A-Queue
405
406
407
408
AR
AR
AR
AR
410
412
414
416
A
A
409
411
413
415
417
Memory
421
422
423
424
E-Queue
E-Queue
E-Queue
E-Queue
426
427
428
ER
ER
ER
ER
431
Arbitration
451
452
453
454
455
456
457
458
461
462
E
X
G
G
T
G
G
X
E
441
442
443
444
445
446
447
448
449

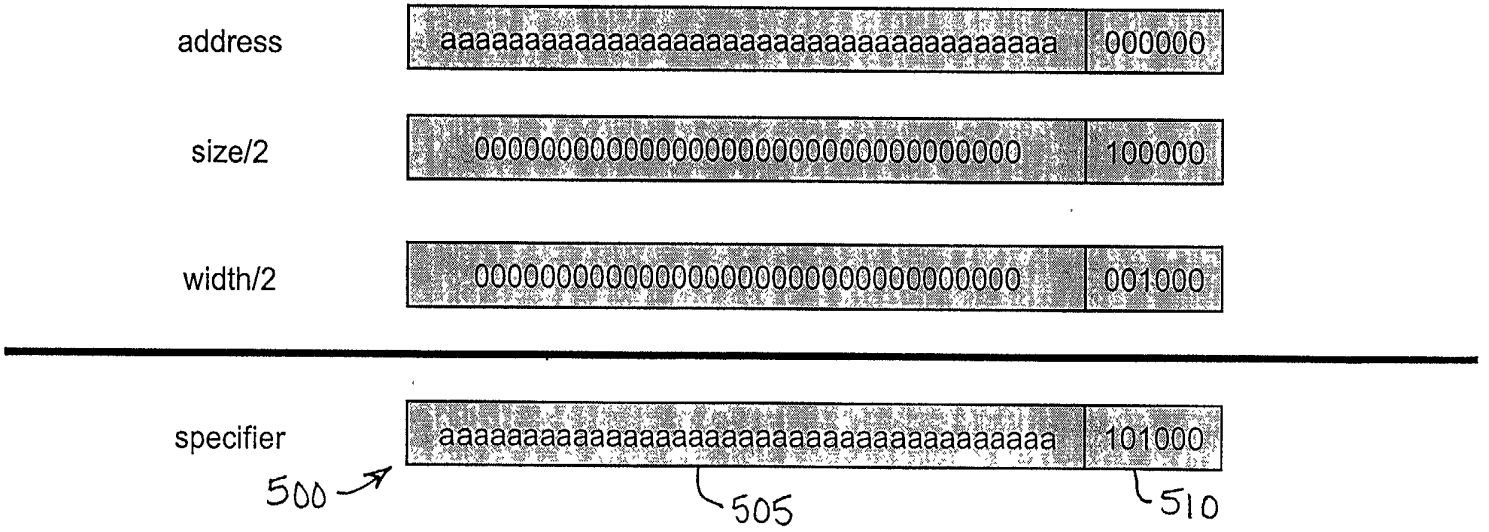
FIG. 4

# Wide operand specifier

■ specifier=address+(size/2)+(width/2)

FIG. 5



# Specifier decoding

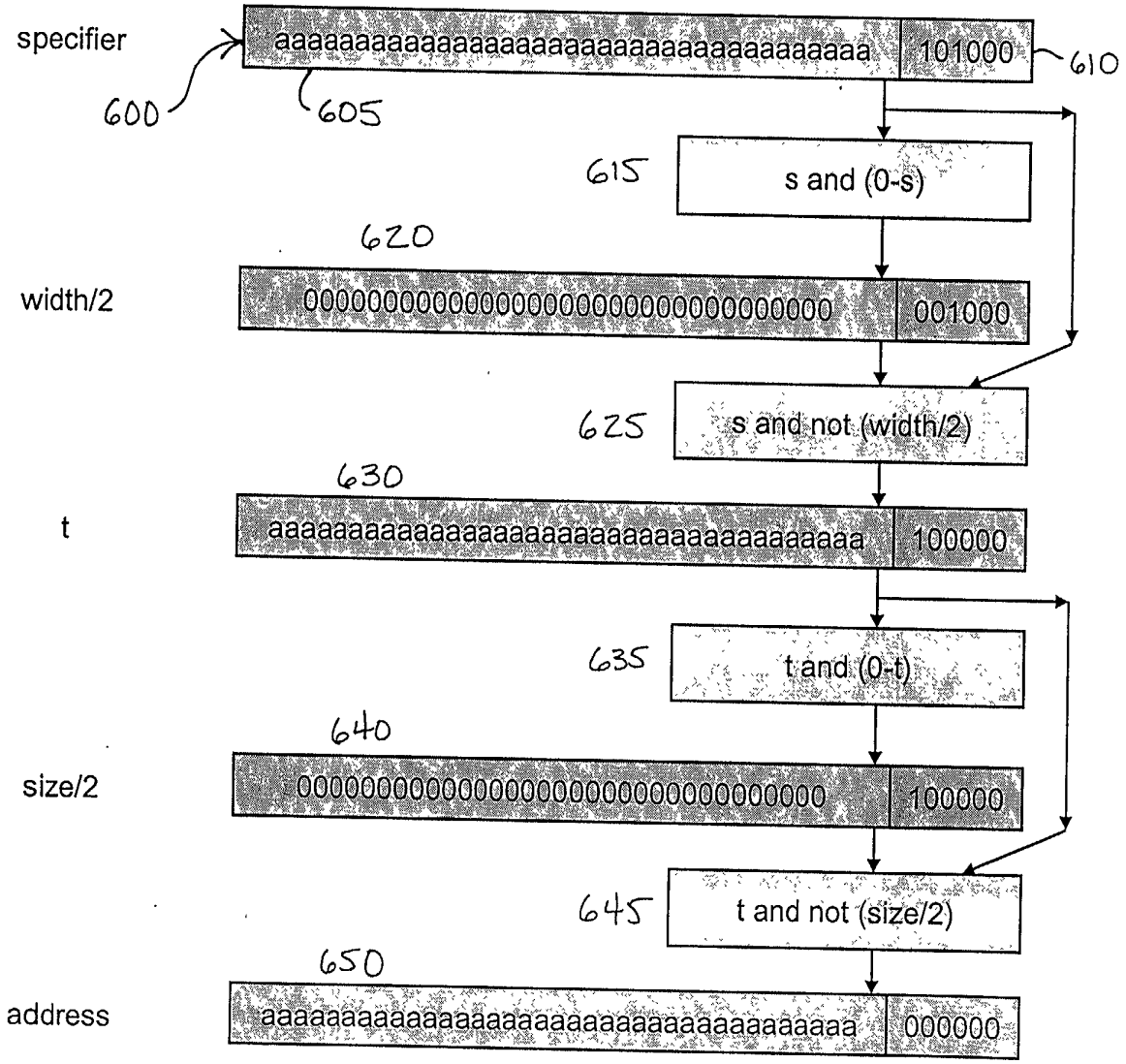



FIG. 6

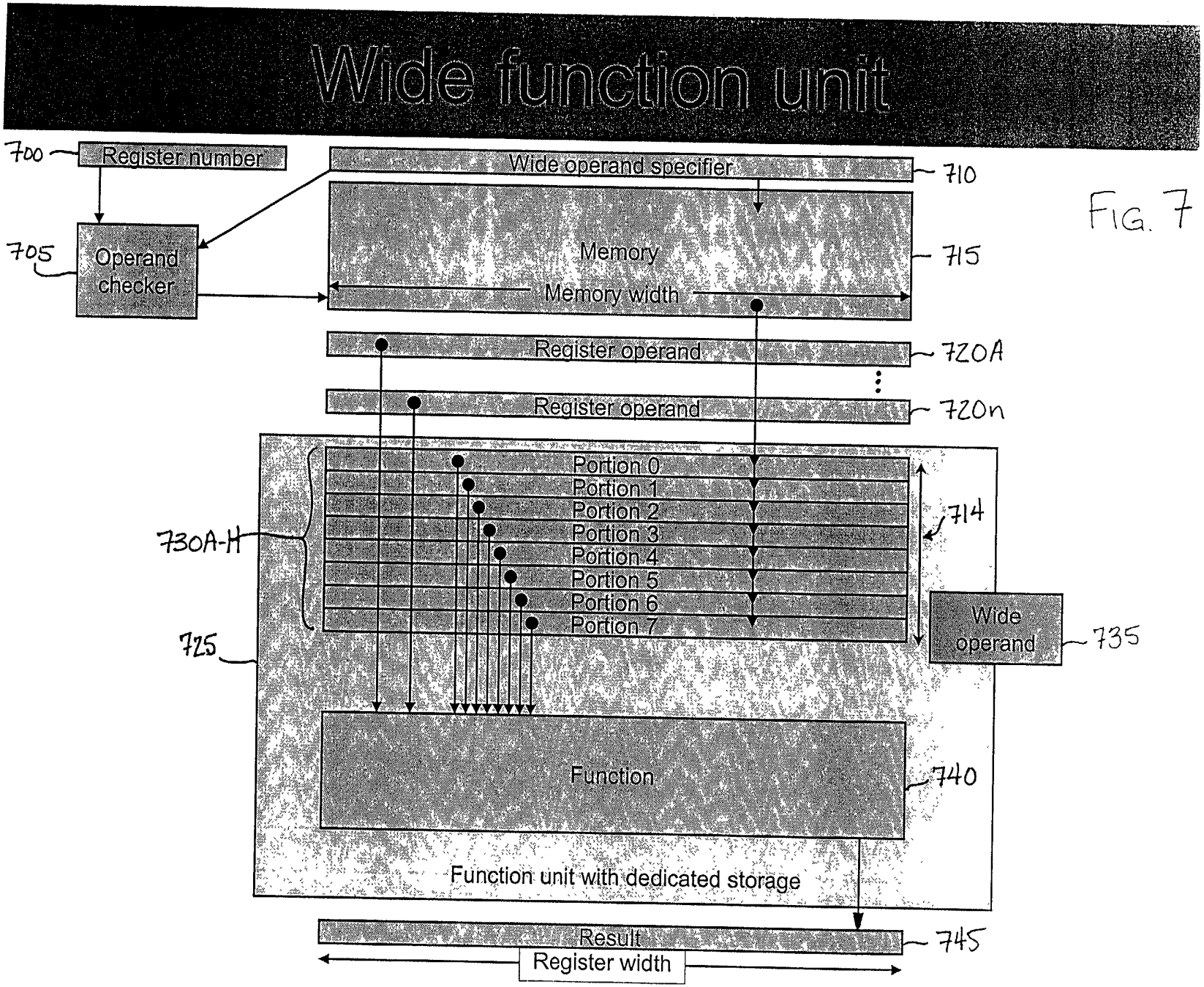

Wide function unit
700
Register number
Wide operand specifier
710
705
Operand checker
Memory
715
Memory width
Register operand
720A
Register operand
720n
730A-H
Portion 0
Portion 1
Portion 2
Portion 3
Portion 4
Portion 5
Portion 6
Portion 7
714
Wide operand
735
725
Function
740
Function unit with dedicated storage
Result
745
Register width

FIG. 7

# Wide MicroCache control

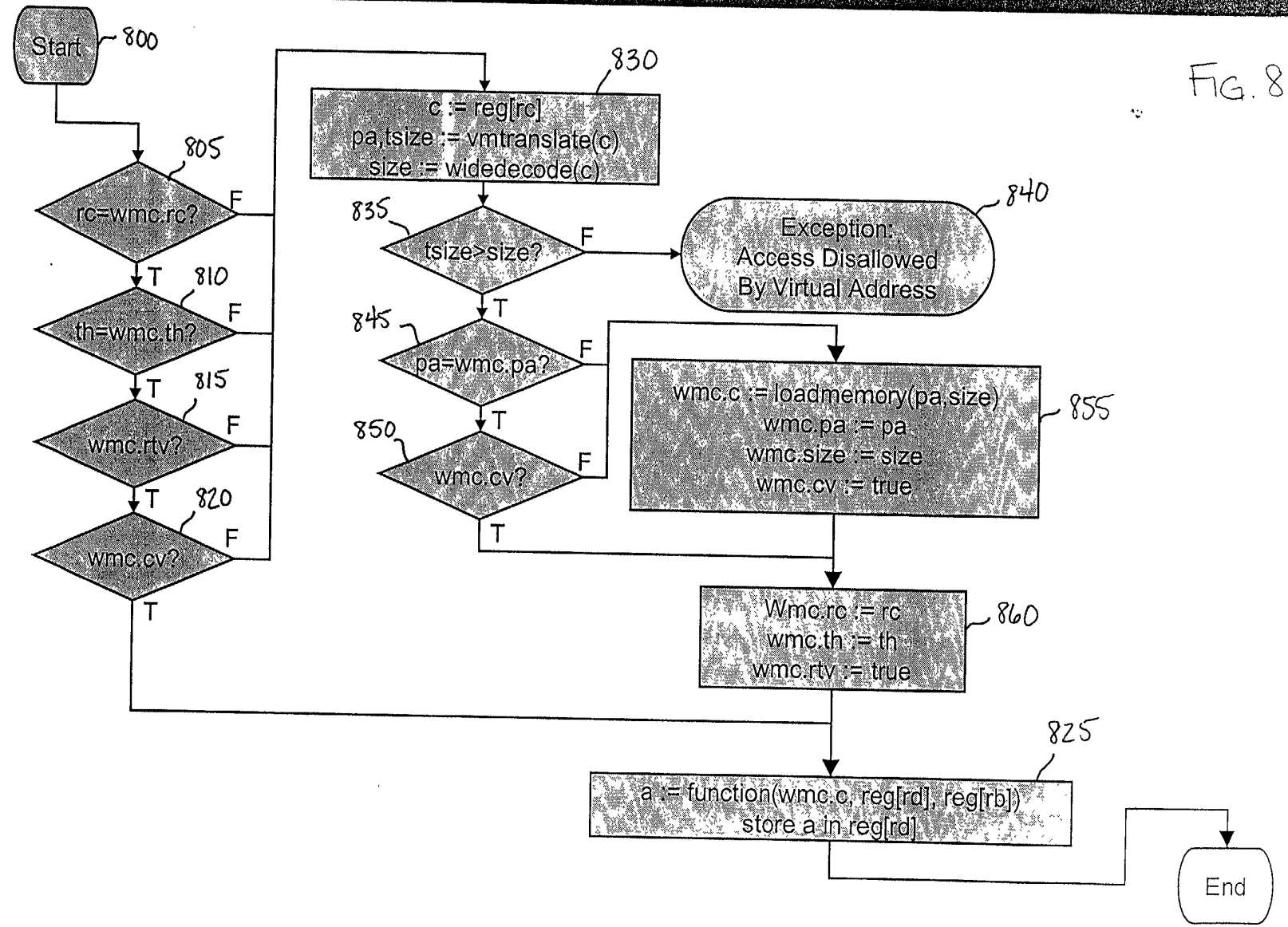
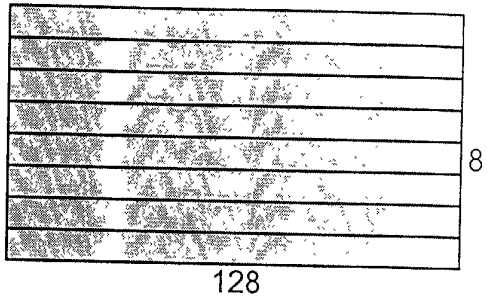



FIG. 8

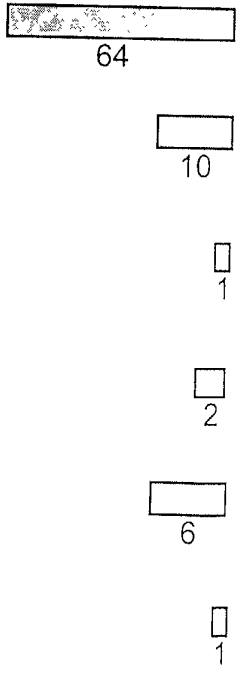
# Wide MicroCache data structures

FIG. 9

- wmc.c contents




- wmc.pa - physical address
- wmc.size - size of contents
- wmc.cv - contents valid
- wmc.th - thread last used
- wmc.reg - register last used
- wmc.rtv - register & thread valid



# Wide MicroCache control (1)

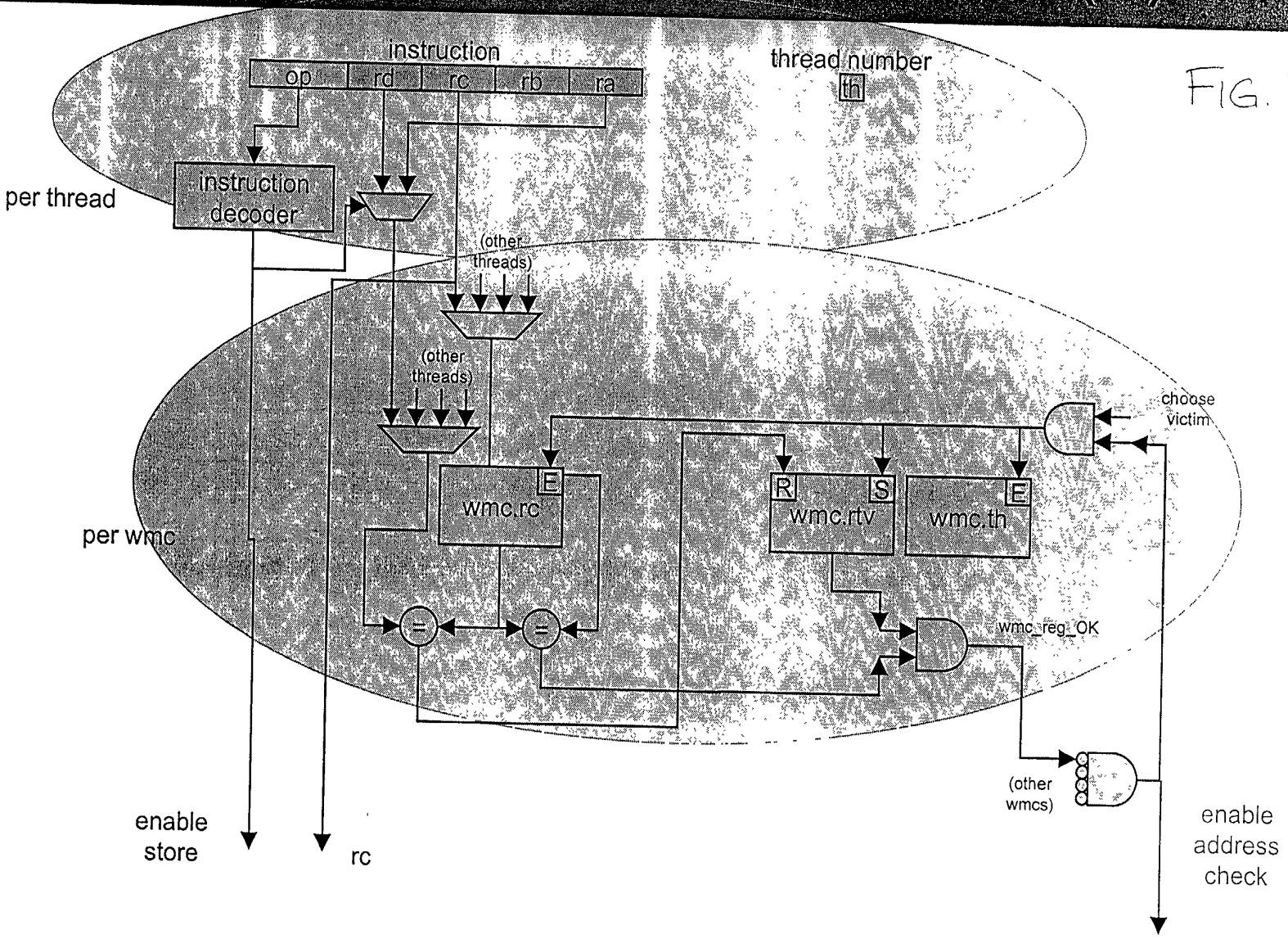



FIG. 10

# Wide MicroCache control (2)

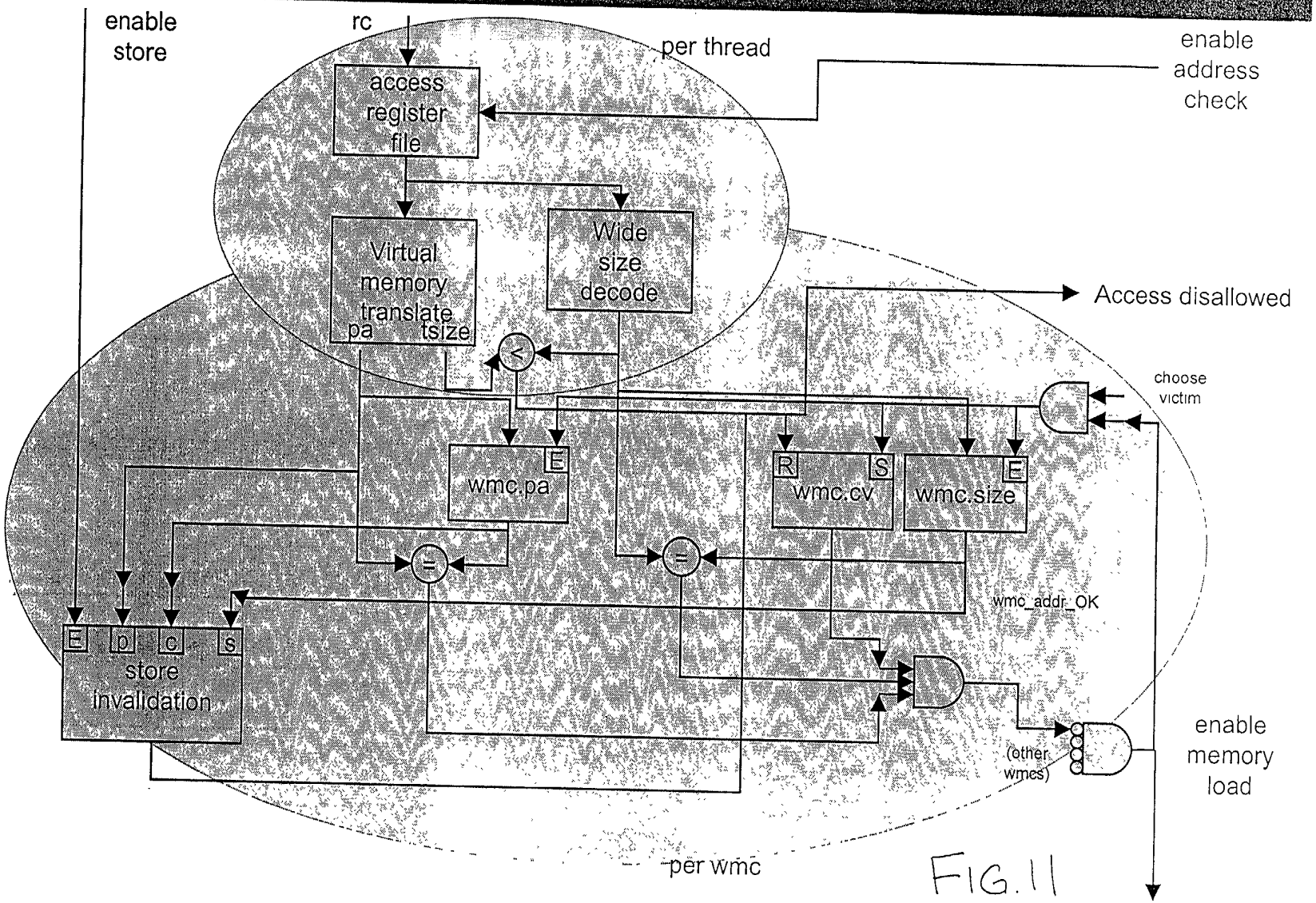



FIG. 11